Revisiting Superpixels for Active Learning in Semantic Segmentation with Realistic Annotation Costs

Lile Cai[1], Xun Xu[1], Jun Hao Liew[2], Chuan Sheng Foo[1]
[1]Institute for Infocomm Research, Singapore
[2]National University of Singapore
{caill, foo_chuan_sheng}@i2r.a-star.edu.sg, alex.xun.xu@gmail.com, liewjunhao@u.nus.edu

Abstract

*State-of-the-art methods for semantic segmentation are based on deep neural networks that are known to be data-hungry. Region-based active learning has shown to be a promising method for reducing data annotation costs. A key design choice for region-based AL is whether to use regularly-shaped regions (*e.g.*, rectangles) or irregularly-shaped region (*e.g.*, superpixels). In this work, we address this question under realistic, click-based measurement of annotation costs. In particular, we revisit the use of superpixels and demonstrate that the inappropriate choice of cost measure (*e.g.*, the percentage of labeled pixels), may cause the effectiveness of the superpixel-based approach to be under-estimated. We benchmark the superpixel-based approach against the traditional "rectangle+polygon"-based approach with annotation cost measured in clicks, and show that the former outperforms on both Cityscapes and PASCAL VOC. We further propose a class-balanced acquisition function to boost the performance of the superpixel-based approach and demonstrate its effectiveness on the evaluation datasets. Our results strongly argue for the use of superpixel-based AL for semantic segmentation and highlight the importance of using realistic annotation costs in evaluating such methods.*

1. Introduction

In recent years, we have witnessed the rapid development and great success brought by deep learning on many computer vision tasks, where the state-of-the-art is dominated by deep learning methods. A key ingredient to the success of deep learning is the availability of large corpora of annotated training data. In practice, however, the annotation cost for labeling such a corpus can be prohibitively expensive, especially for dense prediction tasks like semantic image segmentation where pixel-wise labeling is needed. Active learning (AL) offers one approach to address this annotation burden by selecting only the most informative samples for labeling.

Previous AL methods for semantic image segmentation can be broadly classified into image-based methods [35, 31, 8] and region-based methods [23, 4, 6] according to the granularity of data selection for annotation. Image-based approaches consider an entire image as a sample while region-based approaches divide images into non-overlapping patches and consider each patch as a sample. Further design choices for the region-based approach are the shape and size of the regions chosen for annotation. Previous work suggests that region-based selection outperforms image-based selection due to the increase in data variability [23]; we thus focus on region-based AL in this work.

A fundamental consideration in navigating these design choices for region-based AL is the cost of annotating a sample. Many works have measured annotation cost in terms of the number (or percentage) of labeled pixels [35, 31, 8, 4, 18], which is not reflective of the polygon-based annotation process used in practice [7]. As an alternative, click-based annotation costs have been proposed [23, 6] to better capture the true annotation costs. More concretely, three types of clicks are usually involved in the polygon-based annotation process: 1) polygon clicks for annotating vertices of the polygon enclosing the object-of-interest; 2) intersection clicks for annotating the intersection points between object boundaries and region boundaries; 3) class clicks for assigning a single class label to each segment within the region. The number of polygon and intersection clicks required will likely be higher for regularly-shaped regions (squares, rectangles) that typically do not respect actual object boundaries. These clicks can be reduced (even avoided) if AL is conducted with boundary-preserving regions, such that the annotator only needs to focus on assigning classes for each region.

This consideration of annotation costs motivates the use of (irregularly-shaped) superpixels in region-based AL [18, 30] instead of regularly-shaped squares or rectangles. Superpixel algorithms divide an image into non-

overlapping irregularly-shape regions by grouping perceptually similar pixels together, such that superpixels preserve natural object boundaries well. As a result, most pixels within a superpixel are from the same semantic category. This enables the use of a light-weight annotation scheme where each superpixel is annotated by only one class label that represents the majority of the pixels, reducing the need for polygon and intersection clicks. As illustrated in Fig. 1, annotating approximately the same number of pixels, "rectangle+polygon"-based method requires more than 10 clicks, while the superpixel-based method only requires 1 click.

However, the advantages of superpixel-based approaches for region-based AL remain unclear. Recent work suggests that the advantage of superpixel-based approaches over pixel-based approaches is marginal [18], possibly because pixel-based annotation costs were used in the evaluation. On the other hand, while more realistic click-based annotation costs have been used to benchmark rectangle-based methods [23, 6], comparisons between superpixel-based methods and rectangle-based methods have not been performed. It therefore remains unclear if a superpixel-based approach can indeed reduce annotation cost compared to the traditional "rectangle+polygon" based approach. We address this question in this work, by revisiting the use of superpixels for region-based AL, performing analyses on the effect of region shape and size on region-based AL with more realistic, click-based measurements of annotation costs.

Our contributions can be summarized as follows:

- We revisit the superpixel-based approach for AL in semantic segmentation with realistic click-based annotation cost taken into consideration, and demonstrate its effectiveness over the traditional "rectangle+polygon"-based approach.
- We investigate how region size affects the superpixel-based scheme and the traditional rectangle-based scheme respectively, and show that the former outperforms for a wide range of region sizes.
- We propose a class-balanced acquisition function to further boost the performance of the superpixel-based approach by favoring the selection of informative samples from under-represented object categories.

2. Related Work

2.1. AL for Deep CNNs

Based on the the criteria used to select samples, AL approaches for deep CNNs can be grouped into three categories: uncertainty-based, diversity-based and the hybrid methods. Uncertainty-based approach defines and measures

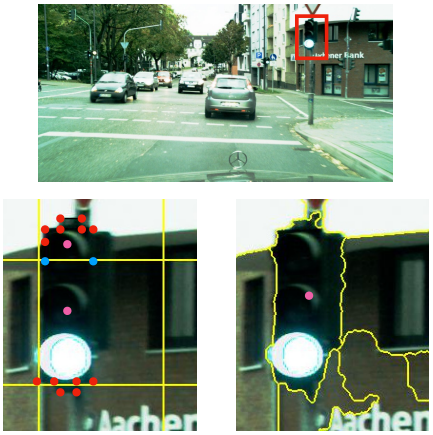


Figure 1: Annotating a traffic light by "rectangle+polygon" based approach (bottom left) *vs*. the superpixel-based approach (bottom right). The former requires quite a few polygon clicks (red dots), intersection clicks (blue dots) and class clicks (pink points), while the latter only requires a class click. If the annotation cost is measured in pixels, the two schemes perform closely, yet when measured in clicks, the latter is much more efficient.

the quantity of uncertainty, and select samples that maximize this quantity. Common measures of uncertainty include entropy [28], variation ratio [13], best-versus-second best margin [17] and the mutual information between predictions and model posterior (BALD) [15]. Instead of using single-model outputs, Gal et al. [14] proposed a Bayesian framework and showed how Monte Carlo (MC) Dropout can be used to obtain posterior uncertainties over network predictions. Beluch et al. [2] used an ensemble of CNNs to derive uncertainty estimates, which outperformed the MC dropout uncertainties. Yoo and In [36] measured uncertainty by loss. Sener and Savarese [26] formulated AL as a core-set selection problem and showed that minimizing the core-set loss is equivalent to the k-centre problem defined over a geometric based similarity function between images. Such approach relies on a feature space that requires extra engineering and suffers from the curse of dimensionality when the feature dimension goes high. Yang et al. [35] used a hybrid framework where samples were first sorted by uncertainty and then Core-Set was applied to the top-k samples to select a subset covering diverse cases. Kuo et al. [19] showed that using the ensemble uncertainty alone performed favourably with the hybrid approach proposed in [35]. In this work we build upon uncertainty-based methods that have been shown to work well for region-based AL.

2.2. AL for CNN-based Semantic Segmentation

AL for semantic segmentation can be classified into image-level approaches [35, 31, 8] and region-level ap-

proaches [23, 4, 6] based on the granularity of sample selection. The former considers the entire image as a sample while the latter divides an image into patches and consider each patch as a sample. In the first category, Yang et al. [35] utilized model predictions and feature descriptors extracted from the trained CNN model to select a set of samples that were most representative and uncertain to annotate. Sinha et al. [31] learned a latent space by a variational auto-encoder (VAE) and an adversarial network, and the prediction score of the adversarial network was used to select samples. Dai et al. [8] also employed VAE to learn a latent space that was used to perform gradient-guided sampling. Among the region-based approaches, methods can be further grouped into regularly-shaped region (*e.g.*, rectangles) based [23, 4, 6] and irregularly-shaped region (*e.g.*, superpixels) based [18, 30] approaches. CEREALS [23] fused entropy with cost estimates to select regions that were informative yet cheap to annotate. Casanova et al. [4] used reinforcement learning to learn the optimal policy for region selection. MetaBox+ [6] selected regions based on prediction quality and cost estimates. Kasarla et al. [18] conduct selection and annotation at the superpixel level and used CRF to refine the superpixel labels. ViewAL [30] proposed a viewpoint entropy formulation that exploited the prediction consistency between different views in multi-view datasets and performed uncertainty computation and selection at superpixel level. The work most related to ours is [18] where superpixel-based selection is used for autonomous vehicle datasets like Cityscapes. However, [18] did not consider realistic annotation costs (clicks), and their results suggest that the advantage of superpixel-based selection without post-processing is marginal. It thus remains unclear if the superpixel-based approach can indeed reduce annotation costs compared to the widely-used "rectangle+polygon"-based approach. Our work addresses this gap.

2.3. Annotation Cost Measurement in AL

Most research in AL assume that the annotation cost is fixed for all samples, and report model accuracy as a function of the number of labeled samples. However, in domains where labeling cost may vary, a reduction in the number of labeled samples does not guarantee a reduction in annotation cost. Settles et al. [27] conduct annotation experiments on four dataset in image and text domains, and showed that annotation costs can vary considerably across instances. For image segmentation, [19] observed that times needed for pixel-wise labeling vary up to 3 orders of magnitude for intracranial hemorrhage segmentation.

AL methods for semantic segmentation are typically benchmarked by the achievable accuracy at some percentage of labeled pixels [35, 31, 8, 4, 18], which may not be a fair comparison as methods require the least amount of labeled pixels may not transfer into the the least amount of annotation time in practice. CEREALS [23] benchmarked various methods with the number of clicks that annotators used to draw a polygon, and revealed that informative regions were more expensive to annotate and selection strategies aiming to select highly-informative regions may not outperform the random baseline significantly if actual annotation cost was considered. CEREALS considered three kinds of clicks: polygon, intersection, and box clicks. MetaBox+ [6] argued that box clicks were not necessarily required with a suitable labeling interface, while class clicks used to select a class label for each polygon should be considered. We consider the same types of clicks as in [6].

2.4. AL for Class Imbalance

Class imbalance refers to the scenario where some classes are represented by significantly fewer instances than others. Real world applications often face this problem as some events or objects are naturally rare in quantity. As most machine learning algorithms optimize the overall classification accuracy, the majority classes may overwhelm training and sacrifice the performance on minority classes. Solutions to this problem include resampling, hard example mining [29] and loss weighting [21]. These methods address the class imbalance at the training stage, while Ertekin et al. [10] showed that AL was capable of solving this problem implicitly by selecting informative samples to annotate at the data collection stage. Explicit handling of class imbalance in AL is often achieved by estimating the pseudo label of an unlabeled sample which decided whether this sample should be preferred in selection. The pseudo label can be obtained by the prediction of the current model or some class-wise similarity measure. Brust et al. [3] addressed the class imbalance problem in object detection by estimating the posterior of each class based on pseudo labels and weighting each instance by the inverse of the posterior to favor under-represented classes. Zhang et al. [37] proposed a similarity-based method for image classification under class imbalance. Kasarla1 et al. [18] represented each category with a feature vector, assigned each pixel to its most similar category and performed selection independently for each class. Similar to [18], we also tackle the class imbalance problem in semantic segmentation. However, instead of discretely assigning annotation budgets to each class, we take a soft weighting strategy based on the pseudo labels of superpixels. This avoids nearest neighbor search in high-dimensional feature space and extra engineering for good features.

2.5. Superpixel Generation

Superpixels are an over-segmentation of an image formed by grouping perceptually similar pixels. It is an established low-level representation of image data that greatly reduces the number of image primitives for subsequent al-

gorithms. Traditional superpixel generation algorithms can be broadly classified into graph-based and clustering-based approaches. Graph-based approaches model an image as a graph where pixels are graph nodes and edges denote affinities between connected pixels. Algorithms falling into this category include Normalized Cuts [25], Felzenszwalb and Huttenlocher (FH) [12] and the entropy rate superpixels (ERS) [22]. Clustering-based approaches group pixels using clustering techniques, which progressively refine an initial clustering of pixels until some criteria are met. Some popular algorithms in this category include SLIC [1], SEEDS [34] and LSC [20]. A more comprehensive survey can be found in [32]. Traditional methods rely on some hand-crafted features to generate superpixels and segmentation ground truth is not used. Recently, SEAL [33] and SSN [16] employed deep CNNs to learn the features for superpixel generation, where the ground truth semantic segmentation label for each pixel is needed to compute the learning loss. In this work, we stick to the traditional methods, *i.e.*, SEEDS, to avoid additional labels for training.

3. Methodology

In this section, we first present an overview of the proposed method. Then, we detail each component of the framework, including superpixel generation (Section 3.2), class-balanced sampling (Section 3.3) and annotation cost measurement (Section 3.4).

3.1. Overall Framework

Given a set of unlabeled images, our method first divides each image into superpixels. Next, we perform class-balanced sampling to select a batch of informative samples, which are then annotated by an oracle. Here, we use the ground truth semantic segmentation label to simulate such annotation process. Instead of the traditional polygon-based labeling, we use a dominant labeling scheme where each superpixel is assigned only a single class label. The model is then retrained using all the data labeled so far and the process is repeated until the annotation budget is exhausted.

3.2. Superpixel Generation

Traditional region-based AL approaches simply divide an image into non-overlapping rectangles, which do not respect natural object boundaries. Superpixels, on the other hand, are image primitives that group similar pixels and preserve object boundary well. In this work, we employ off-the-shelf SEEDS algorithm [34] due to its good performance in ensuring class coherency within each superpixel while maintaining object boundaries and ready-to-use interface (publicly available in OpenCV[1]). In short, SEEDS is a clustering-based superpixel generation algorithm that begins with a uniform partition of image and iteratively refines the results by exchanging neighboring blocks in a coarse-to-fine manner. Note that our proposed pipeline is a universal one and any other superpixels algorithms may be employed.

[1] https://docs.opencv.org/3.4/df/d6c/group_ximgproc_superpixel.html

3.3. Class-Balanced Sampling

Given the pre-computed superpixels, we next describe our strategy for selecting samples for query. AL is typically an iterative process where a batch of samples are annotated at each iteration to train a new model. At iteration t, we denote the model as M_t, unlabeled set as $\mathcal{U}_t$, each sample as s, an acquisition function $a(s, M_t)$ is a function that the AL system uses to query the next sample:

$$s^* = \arg\max_{s \in \mathcal{U}_t} a(s, M_t). \tag{1}$$

The most commonly used acquisition function is based on uncertainty, *e.g.*, entropy [28], variation ratios [13], *etc*. In this work, we employ the uncertainty measure proposed in [17], *i.e.*, Best-versus-Second Best (BvSB) margin, which is less affected by small probability values of unimportant classes. Mathematically, BvSB is defined as the ratio between the posteriors of the two most confident classes:

$$u(x, M_t) = \frac{p(y = c^{sb}|x, M_t)}{p(y = c^{b}|x, M_t)}, \tag{2}$$

where c^{sb} and c^b is the class label for the second largest and the largest posterior probability predicted by M_t, respectively. The uncertainty of region s is then defined to be the average uncertainty of pixels within this region:

$$u(s, M_t) = \frac{\sum_{x \in s} u(x, M_t)}{|\{x : x \in s\}|}. \tag{3}$$

Nevertheless, we notice that in practice, there are many datasets with imbalanced class distributions where simple acquisition function based on uncertainty above is incapable of querying samples from rare object categories. As a result, the performance of these under-represented classes significantly degrade due to insufficient training samples. To overcome this, we propose a simple yet effective strategy to favour samples from the under-represented classes during the selection process. Specifically, we first obtain an estimation of class distribution by assigning a dominant label (the class label of the majority pixels within this region) to each region:

$$\begin{aligned} Do(s) &= \arg\max_{c \in \mathcal{C}} |\{x : l(x) == c \text{ and } x \in s\}|, \\ l(x) &= \arg\max_{c \in \mathcal{C}} p(y = c|x, M_t), \end{aligned} \tag{4}$$

where $\mathcal{C}$ is the set of class labels. This gives the posterior of class distribution as follows:

$$p(cls) = \frac{|\{s : Do(s) == cls\}|}{\sum_{c \in \mathcal{C}} |\{s : Do(s) == c\}|}. \tag{5}$$

We then assign a weighting to the uncertainty measure of s based on the class posterior and propose the following class-balanced acquisition function:

$$a(s, M_t) = u(s, M_t)e^{-p(Do(s))}. \quad (6)$$

Given an annotation budget of K clicks, the algorithm to select a batch of samples is summarized in Algorithm 1. Here $cost(s)$ represents the real annotation cost for sample s, the computation of which is described in Section 3.4. Note that the cost is not used for sample selection as CEREALS [23] and MetaBox+ [6] did, but is used to simulate the actual annotation process where the annotator will stop labeling when the annotation budget is exhausted.

Algorithm 1: Batch-Mode Active Selection

Input : unlabeled set of regions $\mathcal{U}_t$, labeled set of regions $\mathcal{L}_{t-1}$ selected in previous batches, model M_t trained on $\mathcal{L}_{t-1}$, annotation budget of K clicks for batch t

Output: Output selected set of regions $\mathcal{B}_t$

$\mathcal{B}_t = \emptyset$;

$total\_cost = 0$;

while $total\_cost < K$ **do**

  $s^* = \arg\max_{s \in \mathcal{U}_t} a(s, M_t)$;

  $\mathcal{B}_t = \mathcal{B}_t \cup s^*$;

  $\mathcal{U}_t = \mathcal{U}_t \setminus s^*$;

  $total\_cost = total\_cost + cost(s^*)$;

end

3.4. Annotation Cost Measurement

The iterative process of AL terminates when the annotation budget is exhausted. In practical setting, the annotation budget may be measured in hours or expenses. Some previous work proposed to use the amount of labeled pixels as a substitute of the real annotation cost. More recently, CEREAL [23] and MetaBox+ [6] advocate the use of clicks as a more realistic measurement of annotation cost. In this work, we follow [6] to consider three types of clicks that an annotator uses to label an image, the computation of which is detailed as below (refer to Fig.1 for an illustrative example for each click type).

Polygon clicks (c_p): These are the clicks used for delineating the object boundaries. Given a region, c_p for annotating this region is equal to the number of polygon vertices needed for this region.

Class clicks (c_c): These are the clicks that define the class for each annotated polygon. To estimate c_c for a region, we extract the connected components based on its ground truth label map and c_c is equal to the number of connected components.

Intersection clicks (c_i): These are the clicks incurred in the "rectangle+polygon"-based approach and caused by the intersection of the region boundaries and natural object boundaries. A region boundary pixel is considered an intersection point if the ground truth label of this pixel is different from the pixel below (for vertical boundary) or the pixel on the right (for horizontal boundary). c_i of a region is equal to the total number of intersection points on its boundaries.

We consider two annotation schemes used to annotate a segmentation dataset and the involved clicks are discussed as below.

Precise labeling (Pr): This is the traditional polygon-based annotation scheme. To obtain pixel-wise precise labeling, the annotator needs to first draw a polygon that delineates each object segment and then assign a class label to each polygon. Moreover, the annotator needs to indicate the points where natural object boundary intersects with the region boundary. Therefore, this type of labeling involves all the three types of clicks, *i.e.*, $c_p + c_c + c_i$.

Dominant labeling (Do): This is the labeling scheme we employ for the superpixel-based approach, *i.e.*, the entire region is assigned the dominant label as defined in Eq. (4). Such an annotation scheme is of low cost and only incurs c_c. Moreover, we do not apply any post-processing step to refine the assigned dominant labels as done in [18].

4. Experiments

4.1. Experimental Setup

Evaluation Datasets We evaluate on two popular datasets for semantic segmentation: Cityscapes [7] and PASCAL VOC 2012 [11]. Cityscapes contains 19 object categories, with a training/validation/testing split of 2975/500/1525. PASCAL VOC 2012 contains 20 object categories, with 1464 images for training and 1449 images for validation. For both datasets, we perform AL on the training set and evaluate the resulting model on the validation set.

Segmentation Model We employ DeepLabv3+ with Xception-65 backbone [5] as our segmentation model in all the experiments. We follow the default setting by setting the atrous rates to 6, 12 and 18, the output stride as 16 and the decoder output stride as 4.

Fully Supervised Baseline AL methods are usually benchmarked by the amount of annotation used to achieves 95% accuracy of the fully supervised baseline. To train the fully supervised baseline, we use a base learning rate of 0.007 and polynomial learning rate decay policy (*i.e.*, $lr = (1 - \frac{iter}{max\_iter})^{power}$ where $power = 0.9$). The evaluation is run in single-scale on the full-size image. For Cityscapes, we trained for a total of 60k iterations with a batch size of 4 and input size of 769×769. The mIoU of the fully supervised baseline is $76.48\%(0.31\%)$ (mean and standard deviation of 3 runs). For PASCAL VOC 2012, we trained for a

total of 30k iteration with a batch size of 12 and input size of 513×513. The mIoU of the fully supervised baseline is $77.80\%(0.32\%)$.

Batch Training Details For each batch of AL, we select and annotate samples which is equivalent to an annotation budget of 100k clicks for Cityscapes and 10k clicks for PASCAL VOC 2012. The segmentation model is then trained on the combination of newly annotated samples and the samples annotated in previous batches. The training hyper-parameters remain the same as the fully supervised baseline.

Region Generation We consider two types of regions in the experiment:

(A) Rectangles (Rec): The image is uniformly divided into non-overlapping rectangles of size $m \times m$. We fix $m = 32$ in Section 4.3 and investigate the effect of different region sizes in Section 5.

(B) Superpixels (Sp): We employ SEEDS algorithm implemented in OpenCV to divide the image into non-overlapping superpixels. Before applying SEEEDS, we first apply histogram equalization to the image to improve its contrast, followed by converting to HSV color space. We use the following hyperparameters for the SEEDS algorithm: `prior` = 3, `num_levels` = 5, `num_histogram_bins` = 10, and `double_step` is enabled to slightly improve the quality of the superpixels. The number of superpixels is specified in such a way that it is the same as the number of rectangles when dividing the image using the Rec scheme.

4.2. Benchmarking Methods

We benchmark with the following selection strategies: **(A) Random:** This scheme randomly selects a region.

(B) Uncertainty: This scheme selects a region using the uncertainty-based acquisition function defined in Eq. (3).

(C) ClassBal: This scheme uses the class-balanced acquisition function defined in Eq. (6), *i.e.*, uncertainty weighted by the inverse of class posterior for region selection.

4.3. Experimental Results

In this section, we conduct extensive experiments with various combination of selection strategies, region types, and annotation types to study the effect of each component. The results are presented and discussed below.

Cityscapes The AL results on Cityscapes are presented in Fig. 2. Fig. 2a shows our main results with annotation cost measured in clicks, and Fig. 2b plots the same results with clicks converted into percentage of labeled pixels.

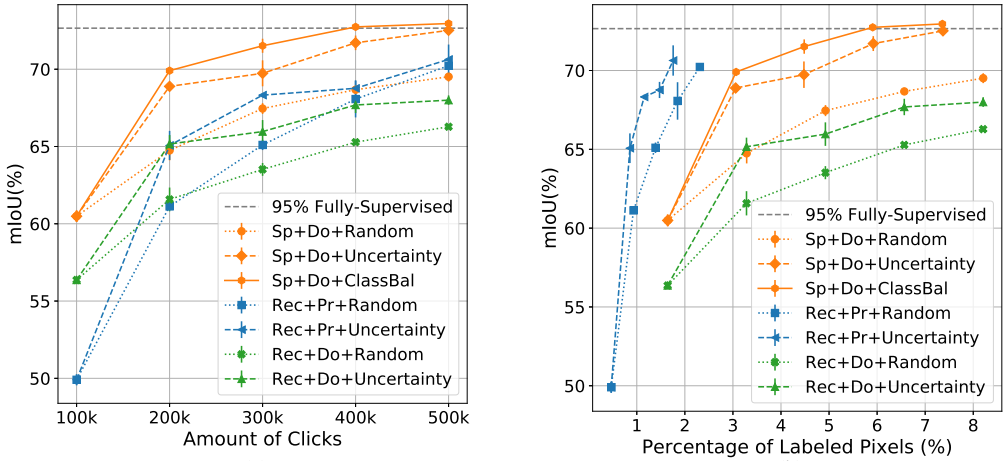
We first observe that Sp+Do outperforms Rec+Pr in Fig. 2a, but loses its advantage in Fig. 2b. This demonstrates that the effectiveness of the superpixel-based approach can only manifests itself significantly when measured in clicks, and the traditional measurement of labeled pixel percentage may under-estimate its performance. We also notice that Rec+Pr outperforms Rec+Do. This indicates that dominant labeling scheme is not effective for rectangular regions, as such regular shape regions do not respect object boundary and can not be effectively represented by the dominant class. Finally, we observe that ClassBal outperforms Uncertainty and Random. This demonstrates that ClassBal can select better samples for model training.

PASCAL VOC 2012 The AL results on PASCAL VOC 2012 are shown in Fig. 3. As PASCAL VOC does not release the click annotation data, we estimate the polygon clicks by fitting a polygon to the segmentation mask: first we run the `alphashape` algorithm[2] with alpha=0.5 to represent the ground truth segment with a concave hull, and then run the RDP algorithm [24, 9] with epsilon=1 to reduce the number of points on the concave hull. The resulting points are used to compute the cost c_p for this dataset (more details in supplementary material).

Similar to Cityscapes, Sp+Do outperforms Rec+Pr in Fig. 3a, but this is not the case in Fig. 3b. This reiterates the importance of benchmarking in realistic annotation cost to properly evaluate the effectiveness of the superpixel-based approach. We also notice that in Fig. 3a, Uncertainty+Rec+Pr cannot beat Random+Rec+Pr, yet in Fig. 3b, the two curves almost overlap. We found that the number of regions selected by Uncertainty can be up to 40% less than Random when using the same amount of clicks. This suggests that regions selected by Uncertainty are more costly to annotate, and Uncertainty-based sampling does not necessarily beat Random sampling when realistic annotation cost is considered. Finally, we observe that ClassBal performs closely to Uncertainty. This may be due to the fact that PASCAL VOC has much more balanced class distribution than Cityscapes and thus the weighting assigned by ClassBal is similar for all the classes.

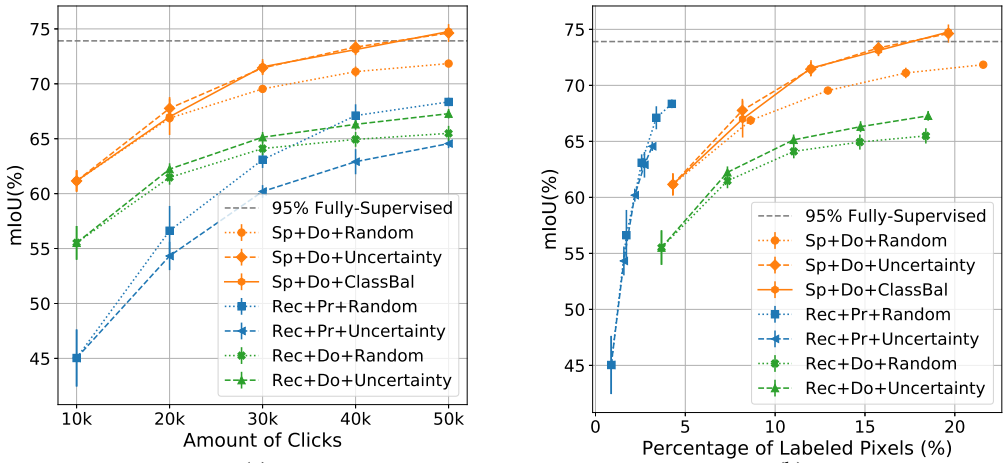
Comparison of Annotation Cost for 95% Accuracy To further demonstrate the effectiveness of the superpixel-based approach, we compare the annotation cost required to obtain 95% accuracy on Cityscapes. We compare with the various methods reported in [6] as the annotation cost is computed in the same way and the same segmentation model is used. As indicated by "Box" in the name, methods in [6] follow the traditional "rectangle+polygon" approach, and "Box+" additionally employ a cost estimation model during sample selection. We convert the absolute clicks into percentage by dividing it by the total number of clicks, which is counted as the number of clicks used to annotate Cityscapes in the original manner, *i.e.*, the entire image is annotated by polygon, and thus is not affected by

[2] https://pypi.org/project/alphashape/



(a) (b)

Figure 2: Active learning results on Cityscapes. We report the mean and standard deviation of 3 runs. (a) Benchmarking at fixed amount of annotation budget measured in clicks. (b) Plot the same results with annotation cost measured in the percentage of labeled pixels.



(a) (b)

Figure 3: Active learning results on PASCAL VOC 2012. We report the mean and standard deviation of 3 runs. (a) Benchmarking at fixed amount of annotation budget measured in clicks. (b) Plot the same results with annotation cost measured in the percentage of labeled pixels.

the number of regions. The results are summarized in Table 1. It can be seen that even Uncertainty+Sp+Do outperforms other methods by a small margin, demonstrating the effectiveness of the proposed Sp+Do scheme. The class-balanced sampling further reduces the annotation cost by 2%.

5. Discussions

Effect of Region Size Region size is an important hyper-parameter for region-based AL. Obviously smaller region size incurs more class clicks (c_c) and intersection clicks (c_i). Table 2 lists the number of clicks for different region sizes on Cityscapes training set. It can be seen that as region becomes smaller, the total number of clicks increases significantly, suggesting that annotating the entire dataset in smaller region is not cost-effective. However, the goal of AL is to maximize model accuracy given a limited annotation budget that is far from enough to annotate the entire dataset. Smaller region size allows the annotation budget to be allocated to label more diverse content. Though the amount of selected pixels is reduced, the increase in pixel diversity can still boost model accuracy. Fig. 4 demonstrates the effect of region size on Cityscapes given a fixed budget of 200k clicks. It can be seen that both the proposed

Table 1: Comparing budgets to obtain 95% accuracy on Cityscapes for different methods.

Random+Rec+Pr	Uncertainty+Rec+Pr	EntropyBox[6]	MetaBox[6]
19.63%	15.70%	19.61%	14.48%
EntropyBox+[6]	Metabox+[6]	Uncertainty+Sp+Do	ClassBal+Sp+Do
10.25%	10.47%	9.81%	**7.85%**

Sp+Do and the traditional Rec+Pr benefit from a smaller region size. This is consistent with the results presented in CEREALS [23]. We also note that Sp+Do outperforms Rec+Pr for a wide range of region sizes from 16×16 to 128×128. When region size goes beyond 128×128, the performance of Sp+Do deteriorates as the superpixel cannot be effectively represented by its dominant label anymore. Pushing the region size to the extreme, *i.e.*, pixel-level selection, is also not optimal, as the same amount of clicks can be used to annotate more pixels if the spatial coherency of neighboring pixels is utilized.

Table 2: The number of clicks for different region sizes on Cityscapes training set. 32×32 corresponds to 2048 regions and 128×128 corresponds to 128 regions per image. Note that c_p is slightly decreasing for smaller region size as we do not count the box corners and polygon clicks on the region boundary have been counted into c_i.

	Polygon(c_p)	Class (c_c)	Intersection (c_i)	Total
Rec+Pr				
32×32	4,170,635	8,952,624	4,987,933	18,111,192
128×128	4,610,236	1,325,930	1,239,878	7,176,044
Image	4,756,857	338,302	0	5,095,159
Sp+Do				
2048	0	6,092,788	0	6,092,800
128	0	380,800	0	380,800

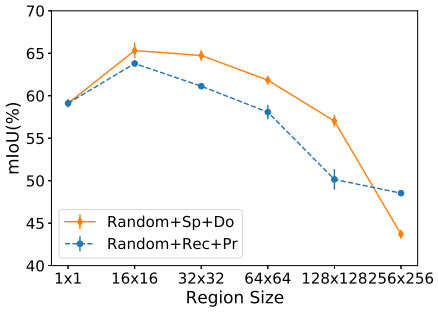


Figure 4: Effect of region size on Cityscapes val subset using the Random baseline.

Effect of Class Balanced Sampling To demonstrate the effect of the proposed class-balanced sampling, we compare the class-wise IoU. Fig. 5a shows the IoU gain of ClassBal+Sp+Do over Uncertainty+Sp+Do at batch 4 (*i.e.*, 400k clicks) and Fig. 5b shows the class distribution of superpixels according to the ground truth dominant label. It can be seen that class-balanced sampling boosts accuracy on less common and especially rare classes, *e.g.*, train, motorcycle, traffic light and traffic sign, while accuracy on popular classes, *e.g.*, road, building, terrain, sky, is largely unaffected.

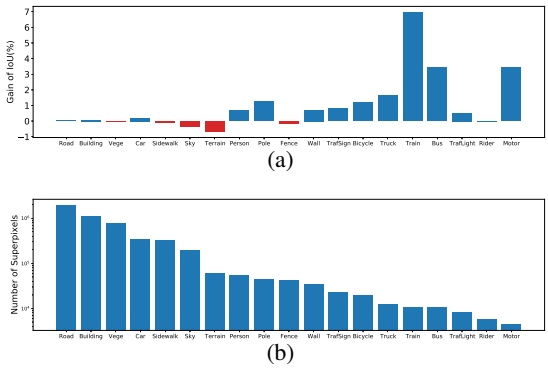


Figure 5: Effect of class-balanced sampling on Cityscapes. (a): the IoU gain of ClassBal+Sp+Do over Uncertainty+Sp+Do at batch 4 (*i.e.*, 400k clicks). (b): the class distribution of superpixels according to the ground truth dominant label.

6. Conclusions

In this work, we revisit the superpixel-based approach for AL in semantic segmentation taking into account more realistic click-based annotation costs. We show that the effectiveness of the superpixel-based approach cannot be properly evaluated by the percentage of labeled pixels, and demonstrate its advantage over the traditional "rectangle+polygon"-based approach on both Cityscapes and PASCAL VOC under realistic cost measurement. We also proposed a class-balanced sampling scheme to further boost the performance of the superpixel-based approach, resulting in a further 25% reduction in annotation cost over the recent Metabox+ method. Our results strongly argue for the use of superpixel-based AL for semantic segmentation and highlight the importance of using realistic annotation costs in evaluating such methods in the future.

Acknowledgment This research is supported by A*STAR under its Learning with Less Data project. The computational work for this article was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg).

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012. 4

[2] William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The power of ensembles for active learning in image classification. In *CVPR*, 2018. 2

[3] Clemens-Alexander Brust, Christoph Käding, and Joachim Denzler. Active learning for deep object detection. *arXiv preprint arXiv:1809.09875*, 2018. 3

[4] Arantxa Casanova, Pedro Pinheiro, Negar Rostamzadeh, and Pal Christopher. Reinforced Active Learning for Image Segmentation. In *ICLR*, 2020. 1, 3

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 5

[6] Pascal Colling, Lutz Roese-Koerner, Hanno Gottschalk, and Matthias Rottmann. Metabox+: A new region based active learning method for semantic segmentation using priority maps. *arXiv preprint arXiv:2010.01884*, 2020. 1, 2, 3, 5, 6, 8

[7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1, 5

[8] Chengliang Dai, Shuo Wang, Yuanhan Mo, Kaichen Zhou, Elsa Angelini, Yike Guo, and Wenjia Bai. Suggestive annotation of brain tumour images with gradient-guided sampling. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 156–165. Springer, 2020. 1, 2, 3

[9] David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10(2):112–122, 1973. 6

[10] Seyda Ertekin, Jian Huang, Leon Bottou, and Lee Giles. Learning on the border: active learning in imbalanced data classification. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 127–136, 2007. 3

[11] Mark Everingham and John Winn. The pascal visual object classes challenge 2012 (voc2012) development kit. *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*, 8, 2011. 5

[12] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004. 4

[13] Linton C Freeman. *Elementary applied statistics: for students in behavioral science*. John Wiley & Sons, 1965. 2, 4

[14] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *ICML*, 2017. 2

[15] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011. 2

[16] Varun Jampani, Deqing Sun, Ming-Yu Liu, Ming-Hsuan Yang, and Jan Kautz. Superpixel sampling networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 352–368, 2018. 4

[17] Ajay J Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2372–2379. IEEE, 2009. 2, 4

[18] Tejaswi Kasarla, Gattigorla Nagendar, Guruprasad M Hegde, Vineeth Balasubramanian, and CV Jawahar. Region-based active learning for efficient labeling in semantic segmentation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1109–1117. IEEE, 2019. 1, 2, 3, 5

[19] Weicheng Kuo, Christian Häne, Esther Yuh, Pratik Mukherjee, and Jitendra Malik. Cost-sensitive active learning for intracranial hemorrhage detection. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 715–723. Springer, 2018. 2, 3

[20] Zhengqin Li and Jiansheng Chen. Superpixel segmentation using linear spectral clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1356–1363, 2015. 4

[21] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 3

[22] Ming-Yu Liu, Oncel Tuzel, Srikumar Ramalingam, and Rama Chellappa. Entropy rate superpixel segmentation. In *CVPR 2011*, pages 2097–2104. IEEE, 2011. 4

[23] Radek Mackowiak, Philip Lenz, Omair Ghori, Ferran Diego, Oliver Lange, and Carsten Rother. CEREALS - cost-effective region-based active learning for semantic segmentation. In *BMVC*, 2018. 1, 2, 3, 5, 8

[24] Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer graphics and image processing*, 1(3):244–256, 1972. 6

[25] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. In *null*, page 10. IEEE, 2003. 4

[26] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, 2018. 2

[27] Burr Settles, Mark Craven, and Lewis Friedland. Active learning with real annotation costs. In *Proceedings of the NIPS workshop on cost-sensitive learning*, pages 1–10. Vancouver, CA:, 2008. 3

[28] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1):3–55, 2001. 2, 4

[29] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on*

computer vision and pattern recognition, pages 761–769, 2016. 3

[30] Yawar Siddiqui, Julien Valentin, and Matthias Nießner. Viewal: Active learning with viewpoint entropy for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9433–9443, 2020. 1, 3

[31] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5972–5981, 2019. 1, 2, 3

[32] David Stutz, Alexander Hermans, and Bastian Leibe. Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 166:1–27, 2018. 4

[33] Wei-Chih Tu, Ming-Yu Liu, Varun Jampani, Deqing Sun, Shao-Yi Chien, Ming-Hsuan Yang, and Jan Kautz. Learning superpixels with segmentation-aware affinity loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 568–576, 2018. 4

[34] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *European conference on computer vision*, pages 13–26. Springer, 2012. 4

[35] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z. Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *Medical Image Computing and Computer Assisted Intervention - MICCAI 2017 - 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III*, pages 399–407, 2017. 1, 2, 3

[36] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *CVPR*, pages 93–102, 2019. 2

[37] Chuanhai Zhang, Wallapak Tavanapong, Gavin Kijkul, Johnny Wong, Piet C de Groen, and JungHwan Oh. Similarity-based active learning for image classification under class imbalance. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1422–1427. IEEE, 2018. 3